本集要点

Python语言的四个特点

Python的五个核心优势

你可能想不到这些著名应用是用Python开发的

找到“我为什么要学Python”的明确职场坐标（金融类）

 本集推荐书单

《利用Python进行数据分析》   作者:  [Wes McKinney](https://book.douban.com/author/1059618/)

 交流方式

本课程提问可至vn.py论坛，[提问](https://www.vnpy.com/forum/forum/17-liang-hua-jiao-yi-ling-ji-chu-ru-men)！‹‹点击

 完整文字版

欢迎来到全新的【量化交易零基础入门系列】——《30天解锁Python量化开发》课程。今天是第一节课，在这一节课里首先要来认识一下python这门编程语言。

### **Python的四个特点**

不管到现在为止你有没有真正尝试过编程，在工作生活里面多多少少听说过这些名词，比如C++、Java，Matlab……这些名词有一个共同点，它们都是一种编程语言。在这么多的编程工具里面， Python有哪些特点呢？总结了四个点：



首先它是由一位叫Guido van Rossum的荷兰程序员在1989年发明的，至今已经有差不多31年的历史。

它是一种面向对象的解释性语言。这句话里面有两个比较关键的名词，一个叫做面向对象object oriented；还有一个叫解释性语言interpreting language。这两个词你现在肯定有一点摸不着头脑，没关系，再过差不多10来节课，相信对这两个名词的概念会有一个非常深入的了解。

Python的设计哲学是优雅、明确和简单。Python的设计更多是为了满足程序员的需求。而不像其他的比如c++这样的语言可能更多是针对CPU的需求来设计的，所以C++能运行的很快。 Python更多针对我们人的需求来设计，可以让程序写得很舒服，写得很快。

Python是一种开源软件，所以首先所有的源代码都提供在网上，可以根据自己的需求任意的去查看、修改和发布，而不用担心它有点会像一个黑盒子。比如Matlab就是一个黑盒子，你看不到里面它到底是怎么实现的。

### **Python的核心优势**

对于我们做量化交易，或者说你想做量化研究数据分析的一些用户来说，Python的核心优势在哪呢？我们把它总结成一句话——易用和功能的完美平衡。平衡在哪里？有这么五个点：



第一点是简单易学。Python的社区时=有一句非常有名的话叫做：人生苦短，我用Python。英文是life is short，use Python。也就是说如果你有条件脱产，不用上学不用上班，三天的时间每天可能8-10个小时，就能够让快速上手Python，开始写出自己的应用代码。如果工作确实有点忙，**每天能抽出1-2个小时，基本上两周、最多三周的时间，也能够上手了**。所以真的是非常容易学，而且不管之前有没有过编程等等方面的经验，只要你愿意学，都能学会。

第二点叫做交互式环境。不像其他的一些编程语言，你可能要写一大段代码，然后做编译，再去运行、看结果。Python所支持的交互式有点像**你打一句命令就给你看一行结果**。所以对于做数据分析、做策略研究的时候，可以大大的提高工作效率。

第三个特点叫做胶水语言。可以轻松的和其他语言开发的功能模块实现对接。比如说像vn.py这样的量化交易框架，我们可以很方便的把上面用Python写的策略和底层基于C++开发的交易接口给对接起来，实现一个完整的量化交易的功能。

第四块叫做应用广泛。不管是做数据分析还是做web开发（就写一个网站或者写一个这种实时的聊天应用）、人工智能机器学习大数据研究，或者说做一个桌面软件，简单的可能就是一个小闹钟，复杂一点的比如做一个像Excel这样的分析图表，或者说做一个个更为复杂的期权波动率交易的一个监控界面，Python都能满足你的需求。

最后一个优势叫做活跃社区。 Python在所有的编程语言里面，它的这么一个用户社区可能是最活跃的。就意味着当你遇到问题的时候，可以在这个社区里面找到很多高质量的扩展类库，或者找到前人已经解决过这些问题的解决方案，直接拿来用就好了。可以帮助我们大大的提高工作效率。

### **那些著名的Python应用**

这里列出了一些Python比较著名的应用。



首先从PPT左上角开始。第一个应用叫做mercurial，它是一个基于Python开发的代码管理工具，什么是代码管理工具？讲另外一个名字可能你更熟一点——github。github背后用的是git这么一个管理工具，而mercurial是和它并列的、世界上用户量最大的两大代码管理工具。

然后是Google APP engine，谷歌语音应用引擎，这个APP engine上开发的就用来开发aAPP的第一的编程语言，也是Python。

右边的这个是YouTube，YouTube它是一个完完全全用Python开发的这么一个网络应用，同时可能也是世界上目前同时在线用户量最大的这么一个Python应用。

然后到第二排，第一个是eve online，对于现在的人来说可能有点陌生，但如果放到10年前，它是第一个可以支持数万游戏玩家同时在线的一个星球大战的游戏。而且同时在线是通过一台服务器就能实现。在当时是一个非常了不起的成就，同样它背后的游戏服务器也是用Python实现的。

然后是openstack，这个是由美国宇航局NASA推出的一个针对云计算的技术框架。现在我们可以看到到处都有各种各样的云：京东云、腾讯云、阿里云等等，这些云计算框架里面，多多少少都借鉴了openstack框架背后的一些技术 open stack，它也是完全用Python开发的这么一套工具。

然后是知乎，是国内最知名的这么一个在线问答交友社区。知乎背后的主要开发语言就是Python。最下面一排Vim，是一个最老的程序员的编辑器， Vim的所有插件开发也同样是用Python来实现的。

然后到BitTorrent，是著名的BT协议，BT这个是用来下载的这么一个工具。 BT生态里面的第一个图形客户端就是用Python来开发的。

最后右下角的是ANACONDA，如果你接触过用Python做数据分析，用Python做一些量化研究的话，我相信ANACONDA可能是你这个知道最有名的这么一个针对科学计算的Python发行版。

所以你可以看到，尽管是一门编程语言，在几乎每一个比较知名的编程语言的应用领域，Python都有很成功的应用案例。

### **Python在金融行业的应用**

接下来我们看两个可能跟我们更相关的。既然来学这个课，主要可能更多关心的还是在金融行业的应用。



第一个来看Python在AQR这么一家全球第二大的量化对冲基金的应用案例。AQR的资金管理规模大概是1592亿美元，这个数字可能是四五年前的，现在相信规模只会更大。在一位前员工叫Wes McKinney，他在AQR工作的时候作为一位量化研究员或者量化分析师要去做很多的金融数据量化分析的工作，但当时手上没有一个特别好的工具，所以他就选择自己一点点摸索开发出了pandas这么一个 Python在数据挖掘领域一重大的开源项目。

可能在今天所有你看到Python在比如说人工智能、机器学习、大数据分析等等这些领域的各种应用，它底层的工具都是pandas。可以说pandas开启了 Python在数据分析这么一个巨大的应用场景里面的一个未来。

Wes McKinney后来也为pandas写了一本书，叫做**《利用Python进行数据分析》**，大家可以直接拿京东或者去淘宝上搜一下这本书可以直接买到，如果你后续考虑在数据分析方面有所建树的话，我强烈地推荐你可以买来先看一下。



直到今天， AQR依旧在大量使用Python进行数据研究和策略开发的工作，并不是说10年前做出来之后现在慢慢淘汰了。直到今天他们大部分的这个数据研究和策略开发的工作都还是在用Python来完成。

AQR在金融机构里面讲叫做买方机构，因为是一个基金。另外一个类型的机构就叫做卖方机构了，在这儿用的例子就是JP Morgan。



JP Morgan是全球最大的银行之一，总资产管理规模高达2.42万亿美元。当然规模上可能略少于我们现在中国的宇宙行（工行），JP Morgan本身是一家Enthought公司的早期用户，同时为numpy等等这些围绕着Python科学计算方面的这些开源项目提供了巨大的帮助。

JP Morgan也是在2008年后把整个IT架构，因为

我们知道2006、2007年爆发的次贷危机，当时在整个次贷危机里面，华尔街投行倒了两个，剩下有一个。剩下的这一个理论上在次贷衍生品上本身是没亏钱的。当然后来整个美国股市暴跌、金融市场流动性干枯，对它也造成了致命的影响。这家公司就是高盛。**高盛为什么能这么聪明地躲开次贷衍生品的亏损？**主要原因就是因为其内部有一套叫做sec language的编程语言（内部开发的），用一句话来形容它的特点：“它是一个面向对象的解释性语言。”

回到第一张PPT里面所讲Python的特点，是不是发现和Python高度类似。确实，这种语言最大的一个帮助就是使得高盛哪怕只是前台最普通的销售——ta压根不是IT也不是 Quant——也能去写一些简单的编程代码，然后很快地针对当时各个trading desk（交易台）上持有的这些次贷衍生品进行一个定价和风险分析。如果需要对冲的就赶紧进行对冲，需要卖掉就赶紧卖掉。这使得高盛在次贷上几乎就没怎么赔钱。也正因为这么一个事实，在2008年后华尔街的各家投行，从美林到JP Morgan等等，都开始把自己的核心的IT架构全面转向用Python来开发了。

当然了C++、Java之类的语言也还在用，但是仅仅来实现性能需求比较高的模块的一些底层代码。这样有什么好处？就像之前说的，一家机构里面不再是最为聪明的IT或者是Quant能写一些程序，就算是前台最普通的、主观的人工交易员、或者说一些甚至于主要负责打电话的销售……他们只要经过很短时间学习，也都可以用Python去写一些小程序来满足自己在交易业务里面的需求。

同时JP Morgan也是开发了一套基于Athena（雅典娜）的这么一个交易和风险管理系统，构建了一套基于Python的金融模型的快速开发环境，使得整个公司金融系统完成了一个比较大的迭代。

这节课的内容就先到这里了，然后更多精华内容请扫码关注我们的社区公众号，下节课再见。